CHARTA LEARNING · LESSON NOTES

Model Risk Management

FRM Part II

1 A story: the model that moved house

- A trusted mortgage model, built on prime borrowers
- Larkfield buys loans to riskier borrowers
- The old model is simply pointed at the new loans
- Defaults turn out three times the forecast
- No bug: the model was used where it did not belong

IN THE LESSON

- Imagine that Larkfield Bank has a mortgage model it trusts.
- It was built on years of data from prime borrowers, people with steady incomes and good credit records, and it has forecast their defaults well.
- Then Larkfield buys a portfolio of loans to riskier borrowers from another lender.
- Nobody builds a new model.
- The old one is simply pointed at the new loans, and its forecasts look reassuringly low.
- Two years later, defaults are three times what the model said.
- Notice that the model's code had no bug, and its maths was fine.
- The damage came from using it where it did not belong.
- That is model risk, and this lesson is about how a bank keeps it under control.

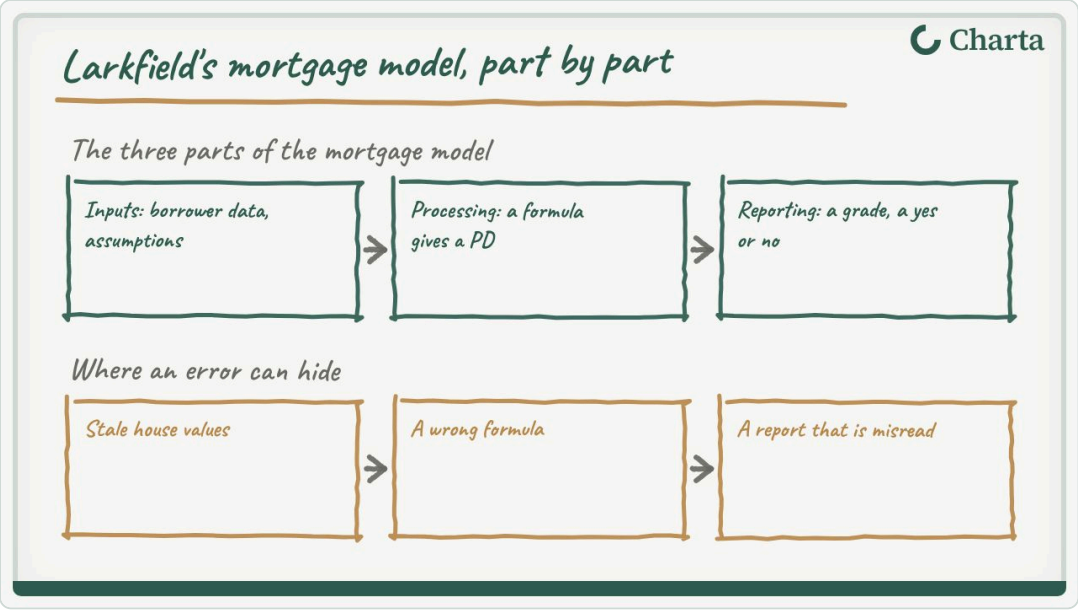
2 What counts as a model

- A quantitative method built on theories and assumptions
- Turns input data into quantitative estimates
- Input component: data and assumptions
- Processing component: the calculation
- Reporting component: information for decisions

IN THE LESSON

- First, what is a model?
- The supervisory guidance defines it as a quantitative method that applies statistical, economic, financial or mathematical theories and assumptions, to turn input data into quantitative estimates.
- Quantitative simply means that it produces numbers.
- Every model has three parts.
- The input component brings in the data and the assumptions.
- The processing component turns those inputs into estimates, by doing the calculation.
- The reporting component translates the estimates into information that people use to make decisions, such as a report, a score or a limit.
- Keep all three in mind, because model risk can enter through any of them, and not only through the calculation in the middle.

3 Larkfield's mortgage model, part by part



IN THE LESSON

- Here is Larkfield's mortgage model laid out in its three parts.
- Look at the top row.
- The input component takes in data on each borrower, such as income, the size of the loan and the value of the house, along with assumptions, for example about future house prices.
- The processing component is a statistical formula that turns those inputs into a probability of default, written P D, the chance that the borrower fails to pay over the next year.
- The reporting component turns each probability into a grade and a yes or no answer for the loan officer.
- Now the bottom row, where an error can hide in each box: stale house values in the inputs, a wrong formula in the processing, or a report that a loan officer misreads.

4 Model risk, defined

- Adverse consequences of decisions based on model outputs
- Financial loss: loans priced too cheaply, capital too low
- Poor business decisions on a flattering forecast
- Reputational damage: trust is lost
- The outputs were incorrect, or misused

IN THE LESSON

- Now the central idea.
- Model risk is the potential for adverse consequences from decisions based on incorrect or misused model outputs.
- The adverse consequences come in three kinds.
- There is financial loss, such as loans priced too cheaply or capital set too low.
- There are poor business decisions, such as entering a market on the strength of a flattering forecast.
- And there is reputational damage, when customers, investors or supervisors lose trust in the bank.
- Notice two words in the definition: incorrect, and misused.
- They point to the two ways in which model risk arises, which we take next.
- Notice too that the harm comes through decisions.
- The risk lies in what people do with the numbers.

5 Two ways model risk arises

Two sources: fundamental error and incorrect use

Source	What goes wrong	Causes
Fundamental error	Inaccurate even as intended	Design, data, assumptions, code
Incorrect use	Sound model, wrong place	Other purpose, portfolio, market
Test	Wrong in its own job: error	Right job, used elsewhere: misuse

IN THE LESSON

- Model risk arises in two ways, and the table sets them out row by row.
- The first is fundamental error.
- The model produces inaccurate output even when it is used exactly as intended.
- The causes are a flawed design, poor data, wrong assumptions or mistakes in the code.
- The second is incorrect or inappropriate use.
- Here the model may be perfectly sound, but it is applied in the wrong place.
- It is used outside its purpose, on a different portfolio, or in market conditions that have changed since it was built.
- The last row gives a test.
- If the output would be wrong even in the job the model was built for, it is an error.
- If the model is right for its own job but used elsewhere, it is misuse.
- Larkfield's story is misuse.

6 Example 1: a sound model in the wrong place

Prime model forecasts 2% defaults; used on $500m of riskier loans; LGD 40%

Expected loss, model view	500m × 2% × 40%	**$4m**
Actual default rate	on the riskier loans	**6%**
Loss actually suffered	500m × 6% × 40%	**$12m**
Shortfall	12 − 4	**$8m**
Actual ÷ forecast	12 ÷ 4	**3 times**

IN THE LESSON

- Let us put numbers on the story.
- Larkfield's prime mortgage model forecasts a default rate of two per cent.
- The bank applies it, unchanged, to five hundred million dollars of riskier loans it has bought.
- Assume that loss given default, the share of a loan lost when the borrower defaults, is forty per cent.
- The model's expected loss is five hundred million times two per cent times forty per cent, which is four million dollars.
- That is what Larkfield provides for and prices in.
- In fact, these borrowers default at six per cent.
- The loss actually suffered is five hundred million times six per cent times forty per cent, twelve million.
- The shortfall is eight million dollars, and the loss is three times the forecast.
- Nothing in the code changed.
- The loss came entirely from misuse.

7 Example 2: what makes model risk bigger

Each driver scored from 1 (low) to 3 (high). A higher total means more model risk

Driver	Mortgage PD	Trading VaR	Branch cash
Complexity	2	3	1
Input uncertainty	2	3	2
Breadth of use	3	2	1
Potential impact	3	3	1
Total	10	11	5

IN THE LESSON

- Model risk is not the same for every model.
- It rises with four drivers: complexity, uncertainty in the inputs, breadth of use, and potential impact.
- Larkfield scores three models on each driver, from one for low to three for high.
- The models are the mortgage default model, the trading desk's value at risk model, and a simple model that forecasts cash needs at branches.
- Complexity scores two, three and one.
- Input uncertainty scores two, three and two.
- Breadth of use, meaning how many decisions rely on the model, scores three, two and one.
- Potential impact scores three, three and one.
- The totals are ten, eleven and five.
- So the value at risk model and the mortgage model deserve the deepest control, and the branch cash model much less.

8 Mistakes to avoid: model risk

- Managed, never eliminated: every model simplifies
- Not only a coding problem
- Sound code can still hide bad assumptions, data, misuse
- Wider use spreads one flaw into many decisions

IN THE LESSON

- Before we move on, here are the points the exam likes to test.
- First, model risk can be managed but never eliminated.
- Every model simplifies the world, so some error always remains, and the goal is to keep that error known and controlled.
- Second, the usual mistake is to treat model risk as only a coding problem.
- A model can be coded perfectly and still rest on a wrong assumption or poor data, or, as at Larkfield, be used for the wrong job.
- Third, the risk grows as a model is used more widely, because one flaw then spreads into many decisions.
- A bank that checks only the code has covered one small corner of the problem.

9 Three pillars of model risk management

- 1. Sound development, implementation and use
- 2. Rigorous validation
- 3. Strong governance, policies and controls
- The pillars are connected: each needs the others

IN THE LESSON

- So how does a bank control model risk?
- An effective model risk management process rests on three connected pillars.
- The first is sound model development, implementation and use: building the model well, putting it into the bank's systems correctly, and using it within its limits.
- The second is rigorous validation, an independent check that the model works as intended.
- The third is strong governance, policies and controls, the framework that sets the rules and makes sure the first two happen.
- The pillars hold each other up.
- Validation struggles with a model nobody documented, and good development is wasted if governance lets an unvalidated model into use.
- We take governance first, then development, then validation.

10 Governance: who sets the rules

Who	Role in governance
Board, senior management	Set appetite; approve policies
Board, senior management	Receive aggregate model risk reports
Policies	Define who does what
Internal audit	Check the whole framework works

IN THE LESSON

- Governance starts at the top, with the board of directors and senior management.
- Together they set the bank's appetite for model risk, meaning how much of it the bank is willing to accept, and they approve the model risk policies.
- They also receive regular reporting on aggregate model risk, the total across all models, not just the latest problem.
- Below them, the policies define who does what, a point we take on the next slide.
- Internal audit is independent of the people who build, use and validate models.
- It checks that the framework actually works, for example that policies are followed and that validation really is independent.
- Audit does not validate the models itself.
- It checks that the system for validating them works.

11 Ownership, control and compliance

Role	Meaning	At Larkfield
Ownership	Business that builds and uses it	Mortgage division
Control	Independent validation, oversight	Validation team
Compliance	Adherence to policy	No model live unvalidated
Accountability	Ultimate responsibility	Board, senior management

IN THE LESSON

- The policies set out three kinds of role.
- Ownership belongs to the business that builds and uses the model.
- At Larkfield, the mortgage lending division owns the mortgage model, and it is responsible for using it properly and knowing its limits.
- Control belongs to independent validation and risk oversight, people who neither build nor use the model and so can judge it fairly.
- At Larkfield that is a model validation team in the risk function.
- Compliance means adherence to policy, for example checking that no model went live without validation.
- Last, accountability.
- Ultimate responsibility for model risk stays with the board and senior management.
- The owner is not the validator, and neither of them replaces the board.

12 The model inventory

- Lists models in use, in development, recently retired
- Records owner, purpose and validation status
- Retired models: their past outputs may still be in use
- Supports assessing model risk in aggregate
- Shows links: one model's output feeds another

IN THE LESSON

- A bank cannot manage models it does not know it has.
- That is why the guidance calls for a comprehensive model inventory.
- It lists every model in use, every model in development, and models recently retired.
- For each one, it records the owner, the purpose, and the validation status, meaning when the model was last validated and with what result.
- Why keep retired models on the list?
- Because decisions and reports made with them may still be in use, and the record shows what replaced them.
- Above all, the inventory lets the bank assess model risk in aggregate, including the links between models, where the output of one model becomes the input of another.

13 Example 3: reading Larkfield's inventory

Larkfield's inventory: 30 of the models in use are Tier 1, the most material

Models in use		**140**
In development		**18**
Retired in the last year		**12**
Total entries	140 + 18 + 12	**170**
Tier 1 overdue for validation	6 ÷ 30	**20%**

The mortgage PD feeds pricing, provisions and capital: one error, three places

IN THE LESSON

- Here is what Larkfield's inventory shows.
- Thirty of its models in use are tier one, the bank's label for its most material models.
- There are one hundred and forty models in use, eighteen in development, and twelve retired in the last year, so the inventory holds one hundred and seventy entries in all.
- Six of the thirty tier one models are overdue for validation.
- That is twenty per cent of the most important models running without a current check, a figure that belongs in the aggregate report to the board.
- Now the links.
- The mortgage model's probability of default feeds the pricing model, the provisioning model and the capital model.
- One error in it reaches three other models, so aggregate risk is more than a list of separate reviews.

14 Effective challenge

- Critical analysis that forces changes
- Objective: independent of builders and users
- Competence: the skill to understand the model
- Influence: the standing to be heard
- Incentives: rewarded for finding problems

IN THE LESSON

- The idea that ties the whole framework together is effective challenge.
- It means critical analysis of a model by objective, informed people who can find its weaknesses and force changes.
- Four conditions make the challenge effective.
- The challengers must be objective, which usually means independent of the people who build and use the model.
- They need competence, the technical skill to understand it.
- They need influence, the standing in the organisation to be heard by senior people.
- And they need incentives that reward finding problems rather than keeping quiet.
- Challenge that cannot force a change is only commentary.
- At Larkfield, nobody with this power ever asked whether a prime model suited riskier borrowers.

15 Example 4: who does what?

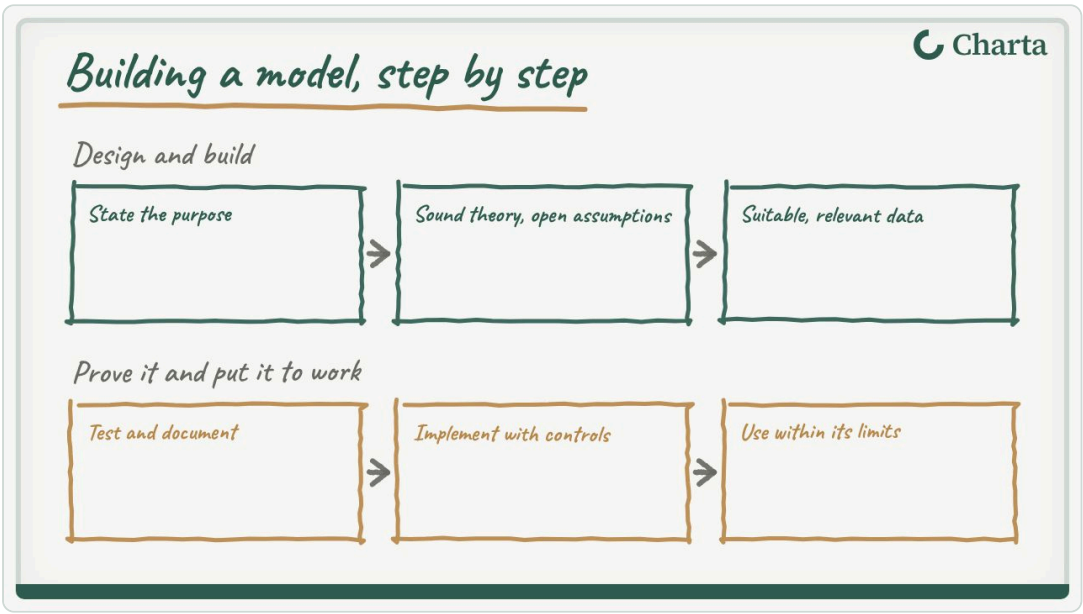
Match each task at Larkfield to the right role

Builds and uses the mortgage model	mortgage division	**Owner**
Tests it independently	validation team	**Control**
Checks no model skipped validation	policy check	**Compliance**
Checks the framework works	internal audit	**Audit**
Ultimately accountable	board, senior management	**Board**

IN THE LESSON

- Exam questions like to blur these roles, so let us sort five tasks at Larkfield.
- Building and using the mortgage model is done by the mortgage division, which is the owner.
- Testing it independently is the job of the validation team, the control role.
- Checking that no model skipped validation is compliance with policy.
- Checking that the whole framework works is internal audit.
- And ultimate accountability stays with the board and senior management.
- Watch for two traps.
- One says that the owner may validate its own model, which removes independence.
- The other says that the validators are accountable for the model, when in fact the business that uses it owns it, and the board answers for the framework as a whole.

16 Building a model, step by step



IN THE LESSON

- Now the first pillar: good practice in building and implementing a model.
- Follow the boxes.
- Everything begins with a clear statement of purpose, which guides both the design and the later use.
- Next comes design on sound theory, with the assumptions stated openly.
- Then data of suitable quality and relevance.
- In the second row, the model is tested and documented, implemented into the bank's systems with controls, and finally used within its limits.
- Larkfield's story went wrong at the first box and the last.
- The prime model's purpose was never written down clearly, so nobody saw that the new portfolio fell outside it, and the model was then used beyond its limits.

17 Purpose, theory and data

- Purpose: what it is for, which portfolio, which decisions
- Design on sound theory and logic, backed by research
- Assumptions and limitations stated openly
- Data of suitable quality and relevance
- Proxy data must be justified and documented

IN THE LESSON

- Let us look more closely at the first three boxes.
- The statement of purpose says what the model is for, which portfolio it covers, and which decisions it supports.
- Design should rest on sound theory and logic, supported by research, so that each relationship in the model has a reason, and not just a good fit to past data.
- The assumptions and limitations must be stated openly, so that users know where the model stops working.
- Data must be of suitable quality and relevant to the purpose.
- Sometimes the right data does not exist, and the builder uses proxy data, meaning data from a similar but different population.
- Any proxy must be justified and documented, because it is a source of error.

18 Testing during development

- Test for accuracy, robustness and stability
- Sensitivity analysis: change inputs, often one at a time
- Stress inputs to extreme but plausible values
- Out-of-sample and out-of-time tests
- A good in-sample fit may only be overfitting

IN THE LESSON

- Testing covers three qualities.
- Accuracy asks whether the estimates are close to reality.
- Robustness asks whether the model still works when inputs are unusual.
- Stability asks whether its results hold steady over time and across samples.
- Three tools do this work.
- Sensitivity analysis changes inputs, often one at a time, and watches how the output moves.
- Stress testing pushes the inputs to extreme but plausible values.
- And out of sample or out of time tests check the model on data it was not built on, either different borrowers or a later period.
- That last test matters because a good fit to the building data may only reflect overfitting, which means the model has learned noise rather than a real pattern.

19 Example 5: the overfitting trap

Two candidate default models, judged on the share of loans misclassified

Model A, 40 inputs: build data	in-sample	**3%**
Model A: a later period	out-of-time	**11%**
Model B, 8 inputs: build data	in-sample	**5%**
Model B: a later period	out-of-time	**6%**
Worsening, A and B	11 − 3 and 6 − 5	**8 and 1 points**

IN THE LESSON

- Larkfield's developers build two candidate models for a new mortgage portfolio, and judge them on the share of loans they misclassify, meaning loans predicted to repay that default, or the reverse.
- Model A uses forty inputs, and misclassifies only three per cent of the loans it was built on.
- On a later period it never saw, it misclassifies eleven per cent.
- Model B uses eight inputs.
- It misclassifies five per cent of the building data, and six per cent of the later period.
- So Model A worsened by eight percentage points, and Model B by only one.
- Model A had learned the noise in its building data.
- The developers choose Model B, even though it looked worse in sample.

20 Example 6: stressing an input

LGD = max(0, 1 − 0.8 × house value ÷ loan). Loan $200,000; house $250,000

Today	1 − 0.8 × 250 ÷ 200	**0%**
House prices fall 20%	1 − 0.8 × 200 ÷ 200	**20%**
House prices fall 30%	1 − 0.8 × 175 ÷ 200	**30%**
Loss on default, after a 30% fall	30% × $200,000	**$60,000**

The base case shows no loss at all; only stressing the input reveals it

IN THE LESSON

- Here is stress testing at work.
- Larkfield's loss given default model assumes that when a borrower defaults, the bank sells the house and recovers eighty per cent of its value.
- So loss given default is one minus point eight times the house value divided by the loan, and never below zero.
- Take a loan of two hundred thousand dollars on a house worth two hundred and fifty thousand.
- Today the formula gives one minus point eight times two hundred and fifty over two hundred, which is zero.
- If house prices fall twenty per cent, the house is worth two hundred thousand, and loss given default is twenty per cent.
- After a thirty per cent fall the house is worth one hundred and seventy five thousand, and loss given default is thirty per cent, a loss of sixty thousand dollars if the borrower defaults.
- The base case alone shows no loss at all; only stressing the input reveals it.

21 Documentation and implementation

- Documentation a knowledgeable third party can follow
- Shows how it works, what it assumes, its limits
- Implementation: verified code
- Controlled integration with other systems
- Change control: record, test, approve before go-live

IN THE LESSON

- Next, documentation.
- It should be detailed enough that a knowledgeable third party, someone competent who did not build the model, could understand how it works, what it assumes and where its limits lie.
- If only the builder understands a model, the bank's knowledge walks out of the door when that person leaves.
- Then implementation, which means putting the model into the bank's production systems.
- The code must be verified, so that it does what the documentation says.
- Integration with other systems must be controlled, so that the right data flows in and the outputs reach the right place.
- And there must be change control: any change to the model is recorded, tested and approved before it goes live.

22 Example 7: when users keep overriding

Loan officers may override the model's grade if they document a reason

Decisions last quarter		**400**
Grades overridden	each documented	**92**
Override rate	92 ÷ 400	**23%**
Policy trigger for review		**10%**
Overrides to a worse grade	80 ÷ 92	**87%**

Rework the model; meanwhile add a margin: PD 2.0% + 0.5 point = 2.5%

IN THE LESSON

- Good practice continues once a model is in use.
- Managers should understand its limitations, and any judgmental override, where a person replaces the model's answer with their own, should be documented.
- At Larkfield, loan officers may override the model's grade if they record a reason.
- Last quarter they made four hundred decisions and overrode ninety two, an override rate of twenty three per cent.
- Larkfield's policy trigger for a review is ten per cent.
- Worse, eighty of the ninety two overrides, about eighty seven per cent, moved borrowers to a worse grade.
- Persistent overrides like these are a sign that the model needs rework.
- Meanwhile, where uncertainty is high, the bank may add conservatism, for example adding half a percentage point to a two per cent default probability, giving two point five per cent.

23 Mistakes to avoid in development

- Testing and documentation start in development
- Validators check the evidence; they do not create it
- A good in-sample fit may only be overfitting
- Persistent overrides signal the model needs rework
- Conservatism helps where uncertainty is high

IN THE LESSON

- Here are the exam angles on development.
- The key one is that testing and documentation start in development, not at validation.
- Validators check the developers' evidence.
- They do not create it after the fact.
- Second, a good in sample fit is no proof of a good model, because it may only reflect overfitting.
- Third, persistent overrides are information, not a nuisance.
- Treat them as a sign that the model needs rework.
- Fourth, adding conservatism, such as a margin on an estimate, is a sensible response where uncertainty is high.
- But it does not remove the need to understand why the model is weak, and to fix it.

24 What validation is

- Does the model perform as intended?
- Are its limitations understood?
- Validators: independent of development and use
- Competent, with influence and incentives to challenge
- US guidance: at least annually, scaled to materiality

IN THE LESSON

- Now the second pillar.
- Model validation is the set of activities that assess whether a model performs as intended, and whether its limitations are understood.
- Notice the two halves.
- A model can pass the first and fail the second, if it works today but nobody knows when it would break.
- Who validates matters as much as how.
- Validators must be independent of both development and use.
- They must be competent, and they must have the influence and the incentives to challenge, which is effective challenge again.
- Reviews are periodic: under the U S guidance at least annually, more often if warranted, and their depth is scaled to the model's materiality.
- So a tier one model gets a far deeper review than the branch cash model.

25 Three core elements of validation

All three are needed: each catches problems the others miss

Element	What it checks
Conceptual soundness	Design, theory, assumptions, data, evidence
Ongoing monitoring	Still fits as products and markets change
Outcomes analysis	Forecasts against actual results

IN THE LESSON

- A strong validation process has three core elements.
- The first is evaluation of conceptual soundness.
- The validator reviews the design, the theory, the assumptions and the data, and the developmental evidence, meaning the tests the developers ran.
- The second is ongoing monitoring, which confirms the model stays appropriate as products and markets change.
- It is how Larkfield should have spotted that its prime model was now running on riskier loans.
- The third is outcomes analysis, which compares the model's forecasts with actual results.
- For a value at risk model this means back-testing: counting the days on which losses exceeded the forecast.
- All three are needed, because each catches problems that the others miss.

26 Ongoing monitoring: three tools

- Process verification: inputs, code, outputs as designed
- Benchmarking: compare with an alternative model or data
- A large unexplained gap is a warning sign
- Sensitivity analysis, repeated in use
- Ongoing: markets change between periodic reviews

IN THE LESSON

- Ongoing monitoring uses three tools.
- Process verification checks that the model runs as designed: that the inputs are complete and accurate, that the code in production is the approved version, and that the outputs reach the reports correctly.
- Benchmarking compares the model's outputs with those of an alternative, such as a simpler model or outside data, and asks why they differ.
- A large gap with no explanation is a warning sign.
- Sensitivity analysis, which we met in development, is repeated in use, to check that the model still responds sensibly as conditions move.
- Monitoring is ongoing, and does not wait for the periodic review, because products and markets can change in between.

27 Example 8: benchmarking the mortgage model

Riskier loans: Larkfield's prime model against outside data on similar borrowers

Model forecast default rate	prime model	**2.0%**
Benchmark default rate	outside data	**5.5%**
Gap	5.5 − 2.0	**3.5 points**
Benchmark ÷ model	5.5 ÷ 2.0	**2.75 times**

An unexplained gap this large: restrict use and investigate

IN THE LESSON

- Here is how monitoring could have saved Larkfield.
- When the riskier loans arrived, the prime model forecast a default rate of two per cent for them.
- A validator benchmarks this against outside data on the default rates of similar borrowers, which shows five point five per cent.
- The gap is three point five percentage points, and the benchmark is two point seven five times the model's figure.
- A gap this large, with no explanation, is a red flag well before any losses arrive.
- The right response is to restrict the model's use on this portfolio and investigate.
- Remember that a benchmark is not the truth either.
- It is a second opinion, and the aim is to explain the difference, not to pick the more comfortable number.

28 Back-testing value at risk

Expected exceptions = (1 − c) × T
Exception: a day when the loss exceeds VaR

c, the confidence level	Confidence level of the VaR
T, the days tested	Number of days tested
Exceptions	Counted from actual results

IN THE LESSON

- Now outcomes analysis with numbers.
- Value at risk, or V a R, is a loss level that a portfolio should exceed only rarely.
- A ninety nine per cent one day V a R should be exceeded on about one day in a hundred.
- If the model is right, the expected number of exceptions equals one minus c, times T.
- An exception is a day when the actual loss is bigger than the V a R.
- Here c is the confidence level of the V a R, and T is the number of days tested.
- The exceptions themselves are counted from actual results.
- Back-testing compares the actual count with the expected one.
- Far too many exceptions mean the model understates risk, and far too few can mean it is too conservative.

29 Example 9: back-testing the trading desk

Larkfield's 99% one-day VaR, back-tested over 250 trading days

Expected exceptions	(1 − 0.99) × 250	**2.5**
Actual exceptions	counted	**9**
Actual ÷ expected	9 ÷ 2.5	**3.6 times**
Chance of 9 or more	if the model is right	**about 0.1%**

One back-test at launch is not validation: it must be repeated

IN THE LESSON

- Larkfield back-tests its trading desk's ninety nine per cent one day V a R over two hundred and fifty trading days.
- The expected number of exceptions is one minus point nine nine, times two hundred and fifty, which is two point five.
- The desk actually had nine exceptions, three point six times the expected number.
- If the model were right, and exceptions were independent from day to day, the chance of nine or more would be only about point one per cent.
- So the model almost certainly understates risk.
- Notice what this check needed: a year of actual results.
- A single back-test run at launch could never have shown it.
- Outcomes analysis has to be repeated, and it sits beside the other two elements rather than replacing them.

30 Difficulties that undermine validation

Difficulty	Why it hurts
Vendor models	Code not disclosed
Scarce skilled staff	Too few able to challenge
Weak independence	Validators report to model users
Poor or short data	Outcomes cannot be tested well
Pressure from the business	Rush to approve, soften findings

IN THE LESSON

- Validation is hard to do well, and several difficulties can undermine it.
- Vendor models, bought from outside firms, often come with undisclosed code, so the validator cannot see inside.
- Skilled staff are scarce, and too few people may be able to challenge a complex model.
- Independence can be weak, for instance when validators report to the managers whose models they review.
- Data may be poor or short, leaving too little history to test outcomes, especially for rare events such as defaults in a downturn.
- And the business may press validators to approve quickly or to soften their findings, because a new model often means new revenue.
- Each difficulty weakens effective challenge in its own way.

31 Mistakes to avoid: validation

- Not a one-time back-test at launch
- Three elements: monitoring always, review at least yearly
- A back-test alone misses broken assumptions
- No independence, no validation
- A vendor model's risk stays with the bank

IN THE LESSON

- To finish the topic, the mistakes to avoid.
- The usual one is equating validation with a one time back-test at launch.
- Validation is a continuing process with three elements: monitoring runs all the time, and the full review is repeated at least annually.
- Second, back-testing alone is not enough.
- A model can pass a back-test in calm markets and still rest on an assumption that breaks in a crisis, which only a review of conceptual soundness would find.
- Third, validation without independence is not validation.
- A developer checking their own work is testing, which is valuable, but it is not effective challenge.
- Fourth, buying a model from a vendor does not hand the model risk to the vendor.
- The bank that uses the model still owns its risk.

32 Try three quick questions

1. 95% one-day VaR, back-tested over 200 days. Expected exceptions?
   Answer: (1 − 0.95) × 200 = 10.
2. Officers overrode 45 of 300 decisions; trigger 10%. What follows?
   Answer: Rate 15% > 10%: review, likely rework.
3. A model passed one back-test at launch. Validated for good?
   Answer: No: soundness and monitoring, independent, yearly.

IN THE LESSON

- Now try three quick questions.
- Pause the video, think, then listen to the answers.
- Question one.
- A ninety five per cent one day V a R is back-tested over two hundred days.
- How many exceptions should we expect if the model is right?
- One minus point nine five is five per cent, and five per cent of two hundred is ten exceptions.
- Question two.
- Loan officers overrode forty five of three hundred model decisions, and the review trigger is ten per cent.
- What follows?
- The override rate is fifteen per cent, above the trigger, so the persistent overrides signal that the model needs review and probably rework.
- Question three.
- A model passed one back-test at launch.
- Is it validated for good?
- No.
- Validation also needs a review of conceptual soundness and ongoing monitoring, done independently and repeated at least annually.